WHAT IS CLAIMED IS:

1. A method for obtaining a cyclic redundancy code for a message, comprising:

separating the message into a plurality of segments;

multiplying a remainder for each segment by a segment-constant based on a generator polynomial to obtain a plurality of segment-remainders;

accumulating the segment-remainders to obtain an accumulated-remainder; and

moduloing the accumulated-remainder by the generator polynomial to obtain the cyclic redundancy code for the message.

2. The method of claim 1, further comprising, moduloing the segments by the generator polynomial to obtain the remainder for each segment.

3. The method of claim 1, further comprising separating the message into three or more segments.

4. The method of claim 1, wherein the cyclic redundancy code is appended to the message and the appended message is transmitted to a receiver.

5. The method of claim 1, wherein cyclic redundancy code indicates the existence of an error in the message.

6. The method of claim 5, wherein integrity of the message is verified if the cyclic redundancy code is zero.

7. The method of claim 5, wherein the integrity of the message invalidated if the cyclic redundancy code is non-zero.

8. The method of claim 1, wherein moduloing includes dividing by the generator polynomial.

9. The method of claim 1, wherein moduloing includes multiplying by a reciprocal-approximator for the generator polynomial.

10. The method of claim 1 wherein the segment-constant for each segment is obtained by moduloing the position of the segment in the message by the generator polynomial.

11. A device for obtaining a cyclic redundancy code for a message, the message separated into a plurality of segments, comprising:

a multiplier to multiply a remainder for each segment by a segment-constant based on a generator polynomial to obtain a plurality of segment-remainders;

an accumulator to accumulate the segment-remainders to obtain an accumulated-remainder; and

a modulo unit to modulo the accumulated-remainder by the generator polynomial to obtain the cyclic redundancy code for the message.

12. The device in claim 11, wherein the device is a network card and the modulo unit includes a plurality of modulo units to modulo the each segment of the message by the generator polynomial to obtain the remainder for each segment.

13. The device in claim 11, further comprising a memory for storing a plurality of segment-constants.

14. The device in claim 11, wherein the segments constants obtain upon receipt of the message.

15. The device in claim 11, wherein the modulo unit divides the accumulated-remainder by the generator polynomial to obtain the cyclic redundancy code.

16. The device in claim 11, wherein the modulo unit multiplies the accumulated-remainder by a reciprocal-approximator for the generator polynomial to obtain the cyclic redundancy code.

17. A method for determining a cyclic redundancy code, comprising:

separating a message into a plurality of segments;

multiplying each segment by a segment-constant based on a generator polynomial to obtain a plurality of segment-remainders;

accumulating the segment-remainders to obtain an accumulated-remainder; and

moduloing the accumulated-remainder by the generator polynomial to obtain the cyclic redundancy code for the message.

18. The method of claim 17, where a degree of a most significant bit of the generator polynomial is greater than a degree of a most significant bit of each segment.

19. The method of claim 17, comprising separating the message into three or more segments.

20. The method of claim 17, wherein the generator polynomial includes a field extender.

21. The method of claim 17, wherein cyclic redundancy code indicates a likelihood of an error in the message.

22. The method of claim 17, wherein each one the plurality of segment-constants is based on the generator polynomial and the position of the segment in the message.

23. A device that obtains a cyclic redundancy code for a message, the message separated into a plurality of segments, comprising:

a multiplier to multiply each segment by a segment-constant to obtain a plurality of segment-remainders;

an accumulator to accumulate the segment-remainders to obtain an accumulated-remainder for the message; and

a modulo unit to modulo the accumulated-remainder by a generator polynomial to obtain the cyclic redundancy code for the message.

24. The device in claim 23, further comprising a memory for storing a plurality of segment-constants.

25. The device in claim 23, wherein the modulo unit divides the accumulated-remainder by the generator polynomial to obtain the cyclic redundancy code.

26. The device in claim 23, wherein the modulo unit multiplies the accumulated-remainder by a reciprocal-approximator for the generator polynomial to obtain the cyclic redundancy code.

27. A method for incrementally updating a cyclic redundancy code, comprising:

subtracting a prior message from an updated message to obtain a difference;

moduloing the difference by a generator polynomial to obtain a remainder; and

adding the remainder to a cyclic redundancy code for the prior message to obtain an updated cyclic redundancy code for the updated message.

28. The method in claim 27, wherein the moduloing includes multiplying the accumulated-remainder by a reciprocal-approximator for the generator polynomial to obtain the remainder.

29. A device which incrementally updates a cyclic redundancy code, comprising:

a subtraction unit to subtract a prior message from an updated message to obtain a difference;

a modulo unit to modulo the difference by a generator polynomial to obtain a remainder; and

an accumulator to add the remainder to a cyclic redundancy code for the prior message to obtain an updated cyclic redundancy code for the updated message.

30. The device in claim 29, wherein the subtraction unit includes exclusive-or logic gates.

31. A method for incrementally updating a cyclic redundancy code for a message, comprising:

subtracting a prior message segment from an updated message segment to obtain a difference-segment;

moduloing the difference-segment by a generator polynomial to obtain a difference segment-remainder;

multiplying the difference segment-remainder by a segment-constant to obtain an expanded segment-remainder;

moduloing the expanded segment-remainder by the generator polynomial to obtain an updated message-remainder; and

adding the updated message-remainder to a cyclic redundancy code for the prior message to obtain an updated cyclic redundancy code for the updated message.

32. The method in claim 31, wherein moduloing includes dividing by the generator polynomial.

33. A device which incrementally updates a cyclic redundancy code for a message, comprising:

a subtraction unit to subtract a prior message segment from an updated message segment to obtain a difference-segment;

a modulo unit to modulo the difference-segment by a polynomial to obtain a difference segment-remainder;

a multiplier to multiply the difference segment-remainder by a segment-constant to obtain an expanded segment-remainder;

a modulo unit to modulo the expanded segment-remainder by the polynomial to obtain an difference-remainder; and

an accumulator to add the difference-remainder to a prior cyclic redundancy code for the prior message to obtain an updated cyclic redundancy code for the updated message.

34. The device in claim 29, wherein the accumulator includes exclusive-or logic gates.

35. An article comprising a machine-readable medium that stores instructions to obtain a cyclic redundancy code for a message, the instructions causing a machine to:

separate the message into a plurality of segments;

multiply a remainder for each segment by a segment-constant based on a generator polynomial to obtain a plurality of segment-remainders;

accumulate the segment-remainders to obtain an accumulated-remainder; and

modulo the accumulated-remainder by the generator polynomial to obtain the cyclic redundancy code for the message.

36. The article of claim 35, further comprising instructions that cause a machine to modulo the segments by the generator polynomial to obtain the remainder for each segment.

37. The article of claim 35, further comprising instructions that cause a machine to verify the integrity of the message if the cyclic redundancy code is zero.

38. The article of claim 35, further comprising instructions that cause a machine to invalidate the integrity of the message if the cyclic redundancy code is non-zero.

39. An article comprising a machine-readable medium that stores instructions to obtain a cyclic redundancy code for a message, the instructions causing a machine to:

separate a message into a plurality of segments;

multiply each segment by a segment-constant based on a generator polynomial to obtain a plurality of segment-remainders;

accumulate the segment-remainders to obtain an accumulated-remainder; and

modulo the accumulated-remainder by the generator polynomial to obtain the cyclic redundancy code for the message.

40. The article of claim 39, further comprising instructions that cause a machine to apply a field extender to the generator polynomial.

41. An article comprising a machine-readable medium that stores instructions to obtain a cyclic redundancy code for a message, the instructions causing a machine to:

subtract a prior message from an updated message to obtain a difference;

modulo the difference by a generator polynomial to obtain a remainder; and

add the remainder to a cyclic redundancy code for the prior message to obtain an updated cyclic redundancy code for the updated message.

42. The article of claim 41, further comprising instructions that cause a machine to obtain the remainder by multiplying the accumulated-remainder by a reciprocal-approximator for the generator polynomial.

43. An article comprising a machine-readable medium that stores instructions to obtain a cyclic redundancy code for a message, the instructions causing a machine to:

subtract a prior message segment from an updated message segment to obtain a difference-segment;

modulo the difference-segment by a generator polynomial to obtain a difference segment-remainder;

multiply the difference segment-remainder by a segment-constant to obtain an expanded segment-remainder;

modulo the expanded segment-remainder by the generator polynomial to obtain an updated message-remainder; and

add the updated message-remainder to a cyclic redundancy code for the prior message to obtain an updated cyclic redundancy code for the updated message.

44. The article of claim 43, further comprising instructions that cause a machine to modulo by division.

45. The article of claim 43, further comprising instructions that cause a machine to modulo by reciprocal approximation.